효율적인 순로코드 발생을 위한 고속 한글 주소검색 시스템 개발

김 경 환[†]·이 석 구[††]·신 미 영[†††]·남 윤 석[††††]

요 약

실제로 사용되는 주소의 분석을 통해 한글주소의 해석방법을 제안하고, 제안한 주소해석 방법을 이용한 주소 검색시스템의 구현에 대하여 서술한다. 주소 상위 및 하위영역의 일치검증을 각각 순차적으로 수행하는 2단계 과정을 통해 최종 배달점에 대한 순로코드를 발생한다. 우편번호와 주소 상위영역 일치검증 단계에서는 우편번호를 이용하여 주소사전에서 검색된 주소단어와 인식된 문자 후보들과의 비교를 통해 우편번호를 검증하게 되며, 주소 상위영역과 주소 하위영역이 분리된다. 주소 상위영역 일치검증 과정의 성능향상을 위해 혼동행렬을 제안하고, 주소 인식결과에 혼동행렬을 적용하여 검증 성공률의 향상을 통해 혼동행렬의 유용성을 확인하였다. 주소 하위영역 검증은 번지정보와 건물명 정보를 이용하여 순로코드를 발생하였다. 부분적으로 완성된 광주와 부산지역의 DPF(Delivery Point File)와 레이블링된 데이터를 이용해 분석 가능한 주소에 대해 높은 정확도를 가지고 순로코드를 발생함을 확인할 수 있었다.

High-Speed Korean Address Searching System for Efficient Delivery Point Code Generation

Gyeonghwan Kim[†] · Seokgoo Lee[††] · Miyoung Shin[†††] · Yunseok Nam[††††]

ABSTRACT

A systematic approach for interpreting Korean addresses based on postal code is presented in this paper. The implementation is focused on producing the final delivery point code from various types of address recognized. There are two stages in the address interpretation : 1) agreement verification between the recognized postal code and upper part of the address and 2) analysis of lower part of the address. In the agreement verification procedure, the recognized postal code is used as the key to the address dictionary and each of the retrieved addresses is compared with the words in the recognized address. As the result, the boundary between the upper part and the lower part is located. The confusion matrix, which is introduced to correct possible mis-recognized characters, is applied to improve the performance of the process. In the procedure for interpreting the lower part address, a delivery code is assigned using the house number and/or the building name. Several rules for the interpretation have been developed based on the real addresses collected. Experiments have been performed to evaluate the proposed approach using addresses collected from Kwangju and Pusan areas.

키워드 : 한글 주소처리(Korean Address interpretation), 순로코드(Delivery point code), 우편 자동화(Postal automation), 혼동행렬(Confusion matrix)

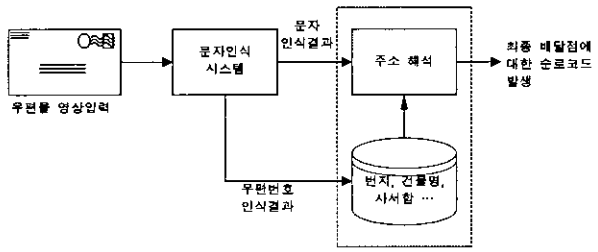
1. 서 론

최근 전자메일, 인터넷, 전화, 팩스 등의 다양한 통신수단의 등장에도 불구하고 문자인식의 가장 큰 응용분야로 분류되는 우정 자동화와 관련된 연구 및 개발 노력은, 오히려 증가하는 우편물의 분류 및 배송과 관련된 처리경비의 절감과 소비자들의 고품질 서비스에 대한 기대의 증가로 인해 꾸준히 증대되어 왔다. 우정 자동화 선진국에서는 이러한 필요성에 대해 일찍부터 인식하고 우정 자동화와 관련된 기술개발에 지속적이며 집중적인 투자를 해왔고, 일부 국가에서는 실제적으로 운용되고 있다[1-3].

우정 자동화 시스템의 개발과 관련된 대표적인 핵심기술에는 고속 영상처리 및 해석[4], 고속 문자인식[5,6]과 최종 목적지 배달순서 자동화[7] 등을 들 수 있는데, 이 중 고속 문자인식과 최종 목적지 배달순서 자동화는 주소해석을 위한 고속 주소 검색시스템의 도움 없이는 높은 처리성능을 기대하기 어렵다.

※ 본 논문은 한국전자통신 연구원의 2000년 위탁과제의 수행을 통한 연구결과 입니다.
† 정 회 원 : 서강대학교 전자공학과 교수
†† 준 회 원 : 삼성중공업 콘트롤시스템 연구원
††† 정 회 원 : 한국전자통신연구원 선임연구원
†††† 정 회 원 : 한국전자통신연구원 우정기술연구부 우정자동화 팀장
논문접수 : 2001년 1월 16일, 심사완료 : 2001년 5월 2일

최종 목적지 배달순서 자동화는 단순히 우편번호 또는 행정단위별 우편물 분류가 아니고, 모든 최종 배달점에 부여된 코드를 이용해 각 집배원의 배달경로까지 고려한 우편물의 자동분류를 의미한다. 우정 자동화 선진국들은 이러한 기술적인 필요성에 대해 일찍부터 인식하고, 이미 모든 최종 배달점에 대한 정보를 체계적으로 수집 및 관리하고 있으며, 이를 우편물 자동분류 시스템과 연동시켜 운용함으로써 우편 배달서비스의 향상과 처리비용 절약효과를 극대화 하고 있다[2, 3]. (그림 1)은 우편번호가 기입된 우편물의 최종 배달점에 대해 순로코드를 발생하는 과정을 보여준다. 문자인식 시스템은 우편물의 주소 영상이 입력되면 수신자 영역에서 우편번호와 주소에 대해 문자인식을 수행하게 된다. 인식된 우편번호를 이용하여 주소 검색시스템에서 관련 주소를 검색하게 되며, 주소 인식결과와 검색된 주소를 이용하여 주소해석을 수행하여 최종 배달점에 대한 순로코드를 발생하게 된다.



(그림 1) 순로코드 발생 시스템

주소 검색시스템은 전국의 주소에 대한 체계적인 구조를 가지게 되는데, 일반적으로 알려진 우편번호에 포함되는 주소체계뿐 아니라 번지정보, 아파트 및 회사명 등을 포함하는 건물명, 사서함 번호 등과 같은 최종 배달점에 대한 정보를 포함하는 대규모 시스템이다. 주소영역을 구성하는 문자들의 고속인식을 위해 인식과정에서 활용 가능한 부분적인 정보를 근거로 해서, 잘못 기재되거나 생략된 주소내용을 보완해 주는 기능을 수행한다. 또한, 인식후보 단어 등과 같은 주소 구성 요소들을 사전형식으로 제공해 줌으로써 인식대상 문자의 수를 제한할 수 있어 인식성능을 정확도와 속도측면에서 모두 크게 향상시키는 역할을 한다[5]. 이러한 검색 시스템의 실질적인 활용을 위해서는 고속 검색능력과 함께 수시로 변하는 최근 주소정보의 반영이 쉬워야 한다[8].

앞에서 언급한 목적을 위한 주소 검색시스템의 설계와 관련된 기존의 연구결과에는 주소를 구성하는 단어들의 인식을 위한 후보단어를 제공하거나 후처리를 통한 검증 목적으로 주소 데이터베이스를 작성하여 활용하는 방법들이 있다[9, 10]. 주소 검색시스템은 후보단어 제공은 물론 순로코드를 발생하는 목적으로 미국의 주소체계에 사용한 연구결과[2, 5]는 있으나, 우리나라의 주소체계와 큰 차이가 있다.

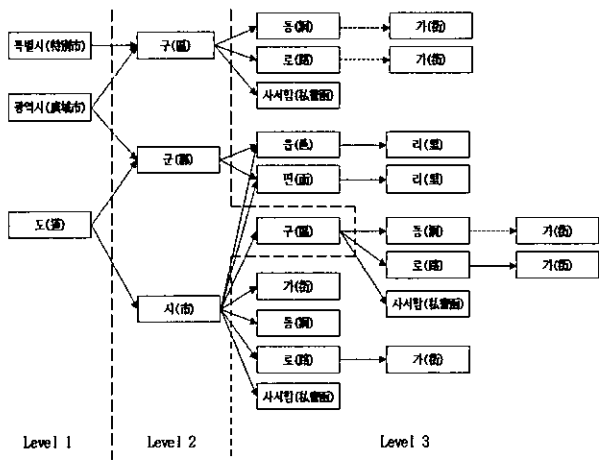
본 논문에서는 우리나라 주소체계의 해석과 실제 사용되고 있는 주소의 분석을 통해 주소의 유형을 순로코드 부여 관점에서 분류하고, 그 결과를 이용해 고속 주소 검색시스템의 구현방안을 제시한다. 이를 위해 일반적으로 알려진 주소체계의 분류는 물론, 실제 사람들이 주소를 기입하는 방식에 대한 분석작업을 함께 수행하였다. 분석결과에 따른 주소해석을 위한 방법을 제안하며, 주소해석 과정에서 주소 및 기타 관련 정보를 효율적으로 활용할 수 있는 주소 데이터베이스의 구조설계 및 검색방법을 제안하고 구현한다. 또한, 입력문서의 왜곡 및 문자모양의 유사성, 인식기술의 제약 등으로 인해 발생하는 문자 오인식 문제를 최소화 하기위한 혼동행렬을 주소해석 과정에서 활용할 수 있는 방법을 제안한다.

본 논문의 구성은 다음과 같다. 제2절에서는 한글주소의 특징을 분류하며, 제3절에서는 우편번호가 기입된 한글주소의 해석방법을 제안한다. 제4절에서는 제안한 주소 해석방법을 이용한 주소 검색시스템의 구현에 대하여 설명하며, 제5절에서는 구현된 주소 검색시스템을 이용한 실험결과 및 분석에 대하여 서술하고, 제6절의 결론을 통해 마무리한다.

2. 한글 주소 체계

2.1 주소 상위영역

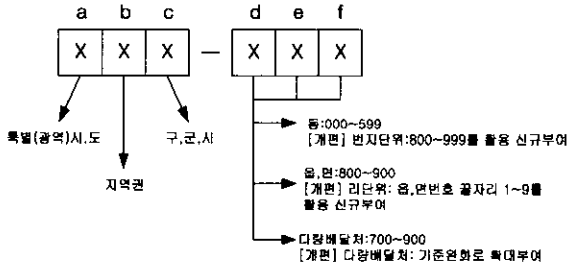
주소 상위영역은 일반적으로 (그림 2)와 같이 계층적인 구조를 갖는다. (그림 2)에서 주소단위 사이에 이어진 실선은 주소를 정의하기 위해서 하위 주소단위가 필수적인 것을 의미한다. 반면, 점선은 하위 주소단위의 사용이 선택적이어서 상위 주소단위 그 자체, 혹은 하위 주소단위를 동반하는 상위 주소단위가 주소 상위영역으로 정의될 수 있음을 의미한다.



(그림 2) 주소 상위영역의 계층적 구조

주소 상위영역은 우편번호가 포함하는 주소 정보와 일치하는데, 이 우편번호 체계는 1970년 7월1일 우체국별로 5자리가 배정되면서 제정되었다. 이후 1988년 2월1일에 1차 개편되어 행정구역별로 6자리의 우편번호가 부여되었으며, 2000년 5월1일에 2차 개편되어 집배원(구)별로 우편번호 6자리가 다시 확대 부여되었다[11].

1차 개편된 우편번호는 6개의 숫자로 구성되었다. 처음 세 개의 숫자는 '도', '구', '군', '시' 등을 구분하는데 사용되고, 나머지 세 개의 숫자는 '동', '로', '가', '읍', '면', '사서함' 등을 구분하는데 사용된다. 반면, 2차 개정에서는, 기존에 '읍', '면', '동' 단위까지 부여된 우편번호를 (그림 3)처럼 '번지'와 '리' 단위까지 세분화 하였다. 개편목적은 우편번호와 집배원 담당구역을 일치시켜 우편물 분류작업의 기계처리율을 대폭 향상시키기 위함이다.



(그림 3) 2차 개편된 우편번호의 체계

2.2 주소 하위영역

주소 하위영역은 주소열에서 주소 상위영역 이후의 주소영역으로 번지, 건물명 등의 구체적인 주소정보를 포함하며 순로코드 발생에 사용된다. 순로코드 발생에 사용되는 정보는 번지, 통/반, 아파트/빌딩/회사명, 사서함, 세대주 등의 형태로 구분될 수 있다. 주소 하위영역의 표현방법은 표준화되어 있지 않기 때문에, 사용자에 따라서 다양한 표현방법이 사용되고 있고, <표 1>에 일반적으로 사용되는 주소형태들을 정리하였다.

〈표 1〉 주소 하위영역에 나타날 수 있는 주소형태

주 소 형 태	표 현 방 법
번 지	488-3, 488번지3, 488-3번지
통, 반	14통1반, 14-1, 14/1
아파트	APT, A.P.T., A, @, 아파트, 아
빌 딩	B/D, BD, BLDG, B, 빌딩
공장단지의 임시번지	3블록5로트, 3BL5L, 3BL/5L,3-5블록

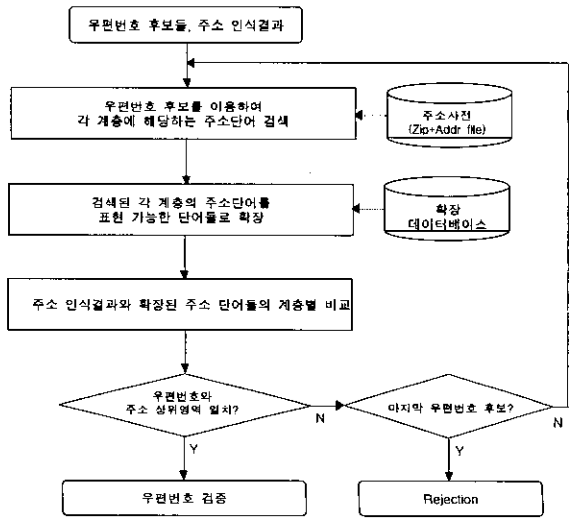
2.3 순로 코드의 발생

주소해석을 통해 순로코드를 발생하기 위해서 주소들은 다음과 같은 조합을 가져야 한다.

- 법정동(행정동 일부)+번지
  법정동 이후에 위치한 번지는 고유성을 가지기 때문에 순로코드를 발생할 수 있다. 법정동이 몇 개의 행정동으로 나누어진 경우, 이 행정동 이후에 오는 번지도 고유성을 가지므로 순로코드를 발생할 수 있다.
- 행정동+통/반+번지
  행정동 이후에 오는 번지정보는 경우에 따라 고유성을 가질 수 없다. 이 경우 행정적 편의를 위해 행정동으로 더욱 세분화한 주소인 '통/반'을 동반한 번지는 고유성을 가져 순로코드를 발생할 수 있다.
- 법정동(행정동)+건물명(아파트,빌딩, 회사 등)
  주소에 번지정보가 없을 경우에 건물명은 번지정보로서 역할을 할 수 있다.
- 사서함+사서함 번호
  사서함 번호는 고유성을 가진다.
- 로+번지, 동+가+번지, 로+가+번지
  번지는 고유성을 가지므로 순로코드를 발생할 수 있다.
- 읍(면)+리+번지
  '리' 이후에 오는 번지는 고유성을 지닌다.
- 읍(면)+리+통/반
  '동'과 '로'에서 '통/반'의 정보는 순로코드를 발생할 수 없으나, '읍'과 '면'에서 '리' 이후에 나오는 '통/반'은 순로코드를 발생할 수 있다.

3. 우편번호가 기입된 한글주소의 해석

제2절에서 언급한 한글주소의 구조를 이용하여 우편번호가 기입된 주소 해석방법을 제안한다. 순로코드를 발생하기 위해 1) 우편번호와 주소 상위영역의 일치검증 단계와 2) 최종 순로코드 발생을 위한 주소 하위영역 검증 단계로 나누어 주소를 해석한다. (그림 4)는 이 과정을 보여주는 주소해석 흐름도이다.



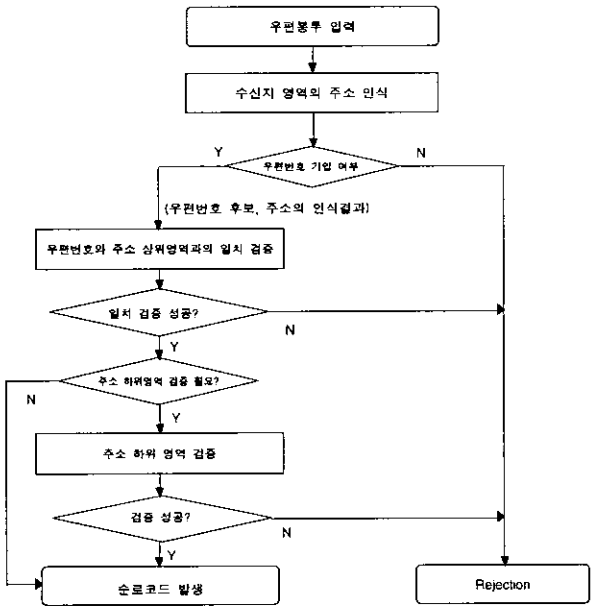
(그림 4) 주소해석 흐름도

3.1 우편번호와 주소 상위영역 일치검증

우편번호와 주소 상위영역 일치검증 단계는 인식된 우편번호 후보들을 이용하여 주소사전에서 주소 단어들을 검색하게 된다. (그림 5)는 우편번호와 주소 상위영역의 일치검증 흐름도이다. (그림 5)에 사용된 주소사전은 주소 상위영역의 계층적 구조정보를 효율적으로 표현하도록 구성한다. 주소사전의 구성에 사용되는 Zip+Addr 파일은 다음과 같이 우편번호에 따른 주소정보를 가지고 있다.

100 - 011 : 서울특별시 : 중구 : 충무로1가
100 - 012 : 서울특별시 : 중구 : 충무로2가
100 - 013 : 서울특별시 : 중구 : 충무로3가
100 - 014 : 서울특별시 : 중구 : 충무로4가
100 - 021 : 서울특별시 : 중구 : 명동1가
100 - 022 : 서울특별시 : 중구 : 명동2가
.....

주소사전은 주소에서 사용될 수 있는 최소단어(예 : 서울특별시는 '서울')만으로 구성하여 사전의 크기를 최소화한다. 실제 주소에서 사용되는 주소단어의 다양한 표현을 위해 '특별시', '시', '구' 등을 포함하는 확장 데이터베이스를 구성하여, 발생 가능한 다양한 표현을 수용하게 된다. 인식된 우편번호에 대해 검색/확장된 주소와 수신자 주소영역의 인식결과의 일치검증을 수행한다.



(그림 5) 우편번호와 주소 상위영역 일치검증 흐름도

3.2 주소 하위영역 검증

주소 하위영역은 주소 상위영역에 비해 매우 다양한 형태로 표현될 수 있다. 따라서, 주소 하위영역의 유형별 분석을 위한 실제 우편영상의 수신자 영역으로부터 labeling된 8,000개 주소의 하위영역을 유형별로 분석한 결과를 <표 2>에 정리하였다. <표 2>에서, case 1-7에 속하는 주소들이 번지정보를 이용해 순로코드를 발생할 수 있는 경우들이며, case 8-10에 속하는 주소들이 건물명을 이용하여 순로코드를 발생시킬 수 있는 경우들인데, 이 두 경우가 전체 분석대상 주소의 76.51%와 20.31%를 각각 차지하고 있음을 알 수 있다. Case 12는 별도의 추가적인 정보(예를 들어 세대주 성명 등)의 활용이 없이는 순로코드를 발생시키기 어려운 주소들로서, 예를 들어 다음과 같은 경우들을 포함한다.

240-043 : 강원 동해시 동해동 5통4반
336-920 : 충남 아산시 송악면 역촌리1구
210-830 : 강원 강릉시 옥계면 주소2리

〈표 2〉 주소 하위영역의 유형분석

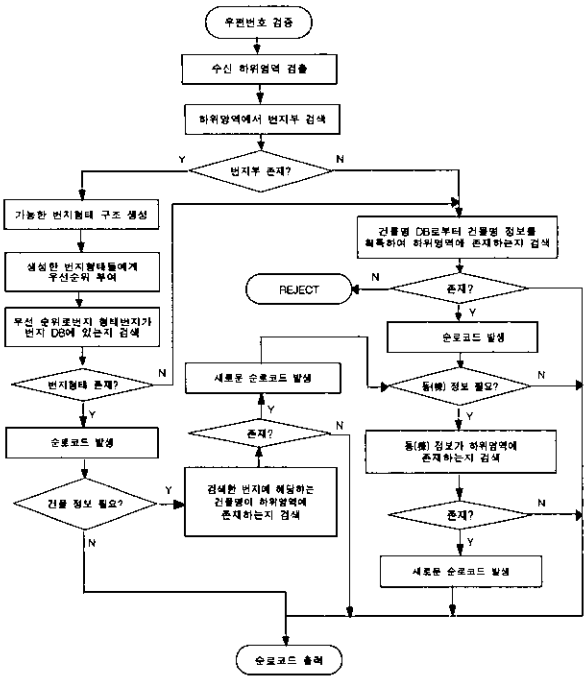
case	주 소 유 형	개수	비율(%)
1	번지	2350	29.38
2	번지+통/반	620	7.75
3	번지+아파트	1810	22.60
4	번지+통/반+아파트	4	0.54
5	번지+건물/회사명/기타(아파트제외)	1260	15.75
6	통/반+번지	23	0.29
7	단지/마을/기타(아파트제외)+번지	16	0.2
8	건물/아파트/상호명/기타(아파트의 경우 동/호 미 기입된 경우)	91	1.14
9	아파트+동+호	1518	18.98
10	블록+건물/아파트/상호명/기타	15	0.19
11	사서함	71	0.89
12	기타	183	2.29

번지정보와 건물명 정보를 이용할 경우 96.82%의 주소에 대하여 순로코드를 발생할 수 있음을 <표 2>를 통해 확인할 수 있다. 따라서, (그림 6)과 같이 번지정보와 건물명 정보를 중심으로 순로코드를 발생하도록 주소 하위영역 검증 단계를 구성하였다.

(그림 6)에서, 번지가 기입된 주소의 경우 번지정보를 이용하여 순로코드를 발생한다. 같은 번지를 사용하는 곳의 경우, 건물명까지 파악되어야 할 경우가 있으며, 아파트의 경우는 동(棟)정보까지 파악되어야 최종 순로코드를 발생할 수 있다. 한편, 건물명 데이터베이스에 등록된 건물명 중, 기입된 건물명과 최대한 일치하는 건물명을 이용하여 순로코드를 발생한다.

주소 하위영역 검증을 위한 데이터베이스의 구성에는 Delivery Point File(DPF)를 사용한다. DPF는 각 우편번호에 대하여 별개의 파일로 구성되며 다음과 같은 구조를 갖는다.

[리이름] : [번지형태] : [본번] : [부번] : [아파트/상호 등의 구분 태그] : [아파트 또는 상호] : [아파트 동수 등의 기타 상세정보]



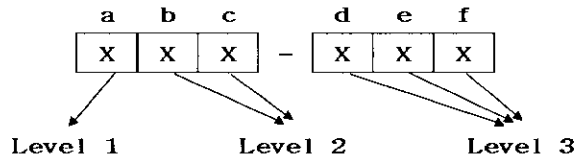
(그림 6) 주소 하위영역 일치검증 흐름도

4. 주소 검색시스템의 구현

4.1 주소 상위영역 검증시스템

우편번호를 이용해 주소사전에서 검색된 주소단어와 인식결과를 비교하여 검증한다. 이 절에서는 우편번호와 주소 상위영역의 관계를 효율적으로 표현할 수 있는 주소사전의 구조 및 검색방법의 구현에 대하여 서술한다. 주소사전은 인식된 우편번호의 유효성검증(validity check) 기능도 포함하게 된다.

우편번호와 주소 상위영역의 관계를 효율적으로 표현하는 주소사전을 구성하기 위해 6자리의 우편번호를 (그림 7)과 같이 3개의 level로 분리하였다. 각 level에 해당하는 부분을 상위영역 주소체계에서 표시하면 (그림 2)의 level 분류와 같다.

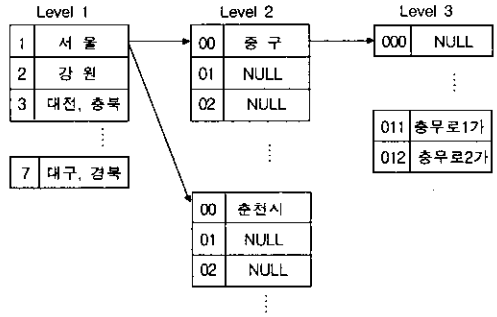


(그림 7) 주소 상위영역의 level (각 level은 그림2의 분류와 일치)

Level 1은 우편번호 중에서 첫 번째 숫자((그림 7)의 'a'에 해당)로서 '특별시', '광역시', '도'를 표시한다. Level 1은 주소 상위영역 중에서 '충청남도', '충남'과 같이 다양한 주소명이 사용되는 곳이다. Zip +Addr 파일이 이러한 다양한 주소명에 대한 변화를 포함하지 않으므로, 주소사전에 주소명 입력 시 프로그램내에서 별도의 구조를 사용하여 다양한 변화를 수용할 수 있도록 한다.

Level 2는 우편번호의 두 번째 숫자와 세 번째 숫자 ((그림 7)의 'b'와 'c'에 각각 해당)로 구성된다. Level 2에서 사용되는 주소명의 마지막 문자는 '시', '군', '구' 중에 하나이다. 따라서 주소사전 구성 시 Zip+Addr 파일에서 주소단어의 마지막 문자 '시', '군', '구'를 확인하여 level 2로 정의한다. 또한 level 2는 '고양시 일산구'와 같이 2개의 주소단어로 구성되는 경우를 고려할 수 있도록 설계하였다. Level 2는 level 1과 연결(link)된다. Level 3는 Zip+Addr 파일에서 level 1과 level 2에서 사용되지 않은 주소단어로 구성되며, level 2와 연결(link)된다.

설계된 주소사전의 고속 검색을 위해 level 1, level 2, level 3는 우편번호를 통해 직접적으로 주소사전에 접근하여 검색하는 구조를 가지며, 전체적인 구조는 (그림 8)과 같다.



(그림 8) 주소사전 구조

- Level 1

Level1은 7개로 구성되며 다음과 같은 구조를 갖는다. (그림 7)의 'a'에 해당하는 숫자를 이용하여 주소사전으로 접근한다.

```
struct _LEVEL1{
        char                    *name ;
        unsigned short          *tag ;
        struct _LEVEL2          *Level2 ;
} *Level1 ;
```

Level1은 다른 level들과 달리 다양한 주소의 표현이 가능하지만, Zip+Addr 파일이 이러한 다양한 표현을 수용하지 않는다. 따라서 다양한 표현은 프로그램 내에서 별도로 처리한다. 아래와 같이 struct의 name에는 주소단어를 입력하고, struct의 tag에 확장자를 입력한다. Level1에서 tag에 의한 확장자는 <표 3>과 같다.

〈표 3〉 Level1의 확장명

확장자	확 장 명
0x00	Null
0x01	특별시, 시
0x02	광역시, 시
0x03	도
0x04	남도, 북도, 도
0x05	남도, 도
0x06	북도, 도

• Level 2

Zip+Addr 파일의 주소단어에서 마지막 문자와 '구', '군', '시'를 비교하여 일치할 경우 level 2로 정의하고, 그 구조는 다음과 같다.

```
struct _LEVEL2{
        char                    *name ;
        unsigned short          *tag ;
        struct _LEVEL3          *Level3 ;
        struct _SUB_LEVEL       *Sub_Level ;
} *Level2 ;
struct _SUB_LEVEL {
        char                    *name ;
        unsigned short          tag ;
        struct _SUB_LEVEL       *Sub_Level ;
} *Sub_Level ;
```

Level 2에 해당하는 주소사전 구성 시 마지막 문자 '구', '군', '시'는 제거되며, struct의 tag에 확장자를 기록한다<표 4>. Level 2의 경우는 '고양시 일산구'와 같이 2개의 주소단어로 구성되어 질 수 있다. 2개의 주소단어로 구성되는 경우는 아래와 같이 struct의 Sub_Level을 이용하여 '고양시'와 '일산구'를 분리하여 주소사전을 구성한다. (그림 7)의 'bc'에 해당하는 숫자를 이용하여 level2 구조로 접근한다

〈표 4〉 Level2와 Level3의 확장명

확장자	확 장 명
0x00	Null
0x01	시
0x02	구
0x03	군
0x04	동
0x05	읍
0x06	면
0x07	가
0x08	로
0x09	리
0x0A	사서함
0x0B	아파트, 아, @, APT, A, 빌딩, B/D, 블록 …

• Level 3

Level 3는 level 1과 level 2에서 사용되지 않은 Zip+Addr 파일의 주소로 구성되고 그 구조는 다음과 같다.

```
struct _LEVEL3{
        char                    *name ;
        char                    *number ;
        unsigned short          tag ;
        unsigned short          Delivery_Poit ;
        unsigned short          low_num, high_num ;
        struct _SUB_LEVEL       *Sub_Level ;
} *Level3 ;
```

'신림13동'과 같은 행정동의 경우 법정동명으로 기입할 경우를 고려하여 '신림'은 struct의 name에 '13'은 number에 기록하고, '동'의 경우는 생략되어 tag에 확장자가 기입된다.

사서함의 경우 아래와 같은 특별한 경우가 존재한다. 이러한 해결을 위해 low_num과 high_num을 이용한다. 또한 low_num과 high_num은 개편된 우편번호에서 번지 범위 기입에 적용할 수 있다.

135 - 600 강남우체국사서함 0001 - 0099
135 - 601 강남우체국사서함 0100 - 0199

Level 3도 level 1과 level 2와 같이 (그림 7)의 'def'를 이용하여 직접적으로 사전구조에 접근을 한다. 사용되지 않는 구조는 NULL로 표현되며, 우편번호의 유효성 판단(validity check)에 사용될 수 있다. (그림 9)는 우편번호를 이용하여 주소사전에서 검색되어지는 주소단어의 예를 보여준다.

우편번호	Level 1	Level 2	Level 3
411-540	경기도 경 기 인천광역시 인천시 인 천	고양시 일산구 고 양 일 산	지영동 지 영
209-817	강원도 강 원	화천군 화 천	상서면 파포리 상 서 파 포 다목리 다 목
140-031	서울특별시 서울시 서 울	용산구 용 산	이촌1동 이촌동 이 촌 동부이촌동 동부이촌

(그림 9) 우편번호를 이용하여 주소사전에서 검색되어지는 주소단어 예

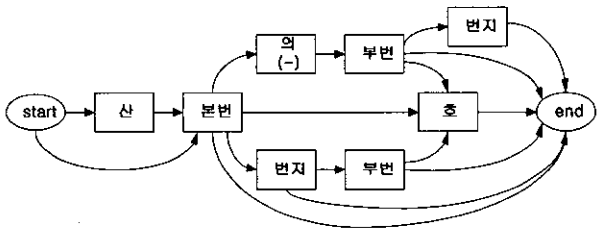
4.2 주소 하위영역 검증 시스템

번지정보와 건물명 정보를 이용하여 순로코드를 발생하도록 구현하였다.

4.2.1 번지부 검색

번지부 검색을 위해 생성할 수 있는 번지부 형태는 (그림 10)과 같고, 조합 가능한 번지부 형태의 예는 다음과 같다.

(1) 산123, 123
(2) 산123호, 123호
(3) 산123-4, 123-4
(4) 산123-4번지, 123-4번지
(5) 산123-4호, 123-4호
(6) 산123번지, 123번지
(7) 산123번지4, 123번지4
(8) 산123번지4호, 123번지4호



(그림 10) 번지부 조합 유형

4.2.2 가능한 번지형태 생성

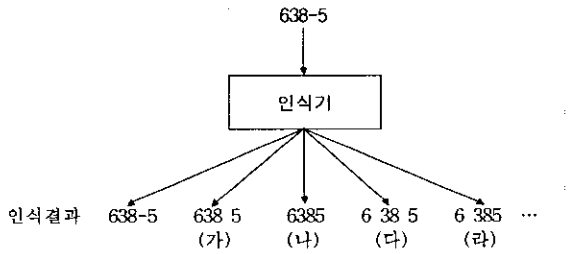
검색된 번지부에 대해 필기자의 습관 또는 인식기의 처리과정 및 성능 등과 같은 실제적인 문제를 고려하기 위하여, 다음과 같은 가정에 의해 가능한 번지형태를 생성한다.

가정1) 본번과 부번을 분리하는 '-'의 경우 낮은 해상도 및 전처리 잡영처리 과정에서의 제거 등으로 인해 인식이 안될 가능성이 있다[12, 13].

가정2) 필기자 습관에 따라 연속된 숫자들 사이에도 띄어쓰기가 발생할 수 있다.

이러한 가정들을 전제로, (그림 11)은 '638-5'라는 주소부가 인식기에서 해석 가능한 유형들을 보여주며, 그림의 각 경우를 설명하면 다음과 같다.

(가) '-'가 띄어쓰기 정보로 처리된 경우
(나) '-'의 크기가 작아 띄어쓰기 정보로 처리지 않았을 경우
(다) 6과 3의 간격이 넓어서 띄어쓰기 정보가 두 곳에서 인식된 경우
(라) 6과 3의 간격이 넓어 띄어쓰기 정보로 처리되지만, '-'의 경우 크기가 작아서 띄어쓰기 정보로 처리되지 못하는 경우



(그림 11) 번지부의 인식유형

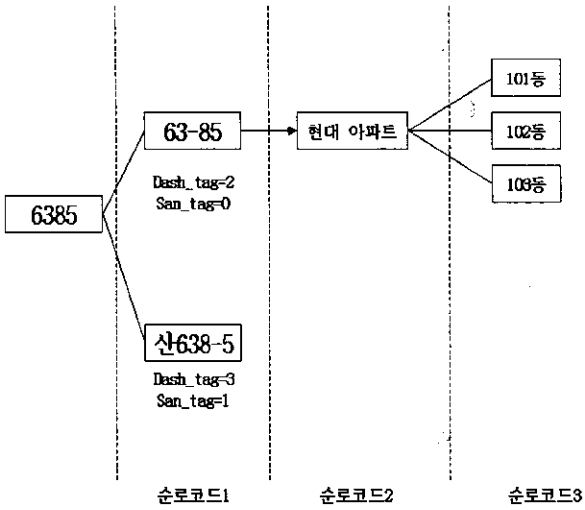
생성 가능한 번지부 형태들을 모두 검색과정을 통해 확인하기 이전에, 다음과 같은 규칙에 의해 우선순위를 부여하고, 우선순위가 높은 주소유형에 대해 우선적으로 번지 데이터베이스에 접근하여 검색하고 유효한 주소일 경우 순로코드를 발생하게 한다.

(1) 주소 하위영역에서 검색된 위치순서에 따라 우선순위 증가
(2) '산', '-', '의', '번지', '호' 등이 포함되어 있으면 우선순위 증가
(3) 번지형태 좌우로 띄어쓰기에 대한 정보가 있으면 우선순위 증가
(4) 번지형태에 띄어쓰기 정보가 포함되면 우선순위 감소
(5) 번지형태가 접한 좌우로 '-', '의', '번지', '호', '통', '/', '동', '산' 등이 있으면 우선순위 감소

4.2.3 주소 하위영역 데이터베이스 구조

번지부는 본번과 부번으로 나눌 수 있다. 본번과 부번은 숫자로 구성되어 있고, 본번과 부번을 구분하기 위해 '-', '의' 등이 사용된다. 그러나 앞에서 설명한 것처럼 인식기의 전처리 과정 등에서 '-'부분이 제거되어 띄어쓰기 정보로 표현될 경우가 있으며, 경우에 따라 띄어쓰기 정보도 획득하지 못할 수 있다.

따라서 본번과 부번을 분리하지 않고 번지 데이터베이스를 구성하며, Dash_tag를 이용하여 본번과 부번을 나눌 수 있는 위치정보를 기록한다. 또한, 산번지 여부는 San_tag를 이용한다. 따라서 생성된 번지형태는 번지부(본번+부번)을 가지고 번지 데이터베이스에 접근한다. 또한, 번지 데이터베이스의 순로코드2를 발생하는 건물명이 연결(link)되어 구성된다. (그림 12)는 번지와 건물명에 의해 순로코드가 결정되는 과정을 보여준다. 즉, 번지정보와 건물명 정보가 공존하는 경우, (그림 12)의 단계를 통해 최종 순로코드가 결정된다.



(그림 12) 건물명에 의해 순로코드가 결정되는 과정 예

4.3 혼동행렬

문자의 유사성 및 인식 기술의 한계 등으로 개별문자의 인식 결과들을 격자구조로 배열하고 이를 검색하는 것만으로는 완벽하게 주소해석을 할 수 없다. 우편번호와 주소 상위영역 검증단계에서는 우편번호로부터 주소 단어들을 제공받음으로 주소열에서 나올 수 있는 단어들을 예측할 수 있다. 따라서, 비슷한 모양 또는 개별 인식 결과를 내는 문자들 사이의 혼동확률을 행렬형태로 정리하여 활용하면 전체적인 주소해석 가능성을 향상시킬 수 있다.

아래의 인식 결과는 행정단위 '동'이 서로 다른 주소영상에서 어떤 신뢰도를 가지고 인식되는지를 7개의 인식후보를 통해 보여 준다.

(a) 동 통 등 둥 돗 퐁 독 : 0.97 0.12 0.01 0.00 0.00 0.00
(b) 듬 튱 픔 쫑 동 픗 꿍 : 0.23 0.00 0.00 0.00 0.00 0.00
(c) 듬 등 용 톱 퉁 픔 풍 : 0.23 0.00 0.00 0.00 0.00 0.00

이 예들에서, 인식기는 '동'을 비슷한 문자들과 혼동하여 인식하고 있음을 보여준다. (a)와 (b)의 경우는 '동'을 인식후보 중에 포함하고 있어 격자구조 탐색만으로 '동'을 검색할 수 있지만 (c)의 경우는 '동'의 인식 결과임에도 불구하고, 7개의 인식 결과를 이용해 '동'을 인식할 수는 없고 결국 주소해석과정에서 거절(reject)된다. 이러한 문제를 해결하기 위해 문자 모양의 유사성과 인식기의 인식특성을 반영한 혼동행렬(confusion matrix)을 사용한다. 문자 '동'이 문자 '듬'과 혼동되어 인식될 가능성이 크다면 그 상관관계를 혼동행렬에 반영하여, 상관관계에 따라 격자구조의 검색에서 활용될 수 있도록 한다.

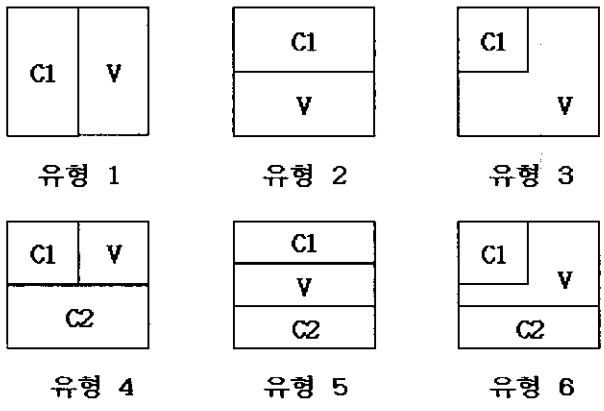
(그림 13) 혼동행렬의 적용(158-075 : 서울 양천구 신정5동)

(그림 13)은 혼동행렬을 적용하여 '서울 양천구 신정5동'을 인식한 결과를 보여주고 있다. 현재 구현된 인식기는 (그림 13)과 같이 문자에 대해 7개의 인식된 문자후보와 신뢰값을 넘겨주게 된다. PRED/SUCC는 인식된 문자의 위치정보이다. '신정5동'의 '정'의 경우는 7개의 문자후보에서 검색을 하지 못하게 되어 검증에 실패하게 된다. 따라서 '징'에 대해 혼동행렬을 적용하여 '신정5동'을 가검증하게 되며, 다른 계층들의 검증 결과에 의해 '징'에 대한 혼동행렬 적용의 타당성이 결정되어 진다.

4.3.1 혼동행렬(confusion matrix)의 구현

한글의 유형은 모음의 형태와 종성의 유무에 따라 (그림 14)와 같이 6가지로 나누어 진다[14, 15].

혼동행렬은 주소 상위영역에서 사용하는 문자들을 문자의 구조에 따라 6개의 유형으로 분리한 후, 문자 영상들의 인식결과를 이용하여 각 유형에 대해 초성, 중성, 종성에 대해 구성한다. <표 5>는 위와 같은 방법으로 구현된 유형 1의 중성 혼동행렬의 예이다.



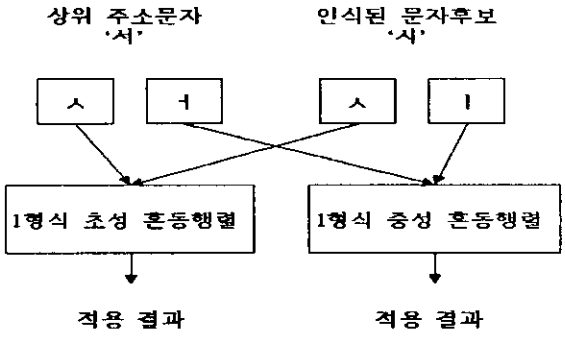
(그림 14) 한글의 6가지 유형 (C1 : 초성, V : 중성, C2 : 종성)

〈표 5〉 유형 1의 중성혼동 행렬

	ㅏ	ㅐ	ㅑ	ㅒ	ㅓ	ㅔ	ㅕ	ㅖ	ㅣ	ㅝ	ㅟ	ㅘ
ㅏ	○	×	×	×	×	×	×	×	○	×	×	○
ㅐ	×	○	×	×	×	○	×	○	×	×	×	×
ㅑ	×	×	○	×	×	×	×	×	×	×	×	×
ㅒ	×	×	×	○	×	×	×	×	×	×	×	×
ㅓ	×	×	×	×	○	×	○	×	○	×	×	×
ㅔ	×	○	×	×	×	○	×	○	○	×	×	×
ㅕ	×	×	×	×	○	×	○	×	○	×	×	×
ㅖ	×	○	×	×	×	○	×	○	○	×	×	×
ㅣ	×	×	×	×	×	×	×	×	○	○	○	×

4.3.2 혼동행렬의 적용

혼동행렬의 적용은 상위 주소문자의 각 자소들이 혼동될 수 있는 자소 정보를 이용한다. (그림 15)를 보면, 유형 1의 '서'의 경우 'ㅅ'과 'ㅓ'로 초성과 중성으로 나눌 수 있다. 인식된 문자 후보 '시'가 입력될 경우 '시'는 'ㅅ' 과 'ㅣ'로 나뉜다. 각각의 자소들을 유형 1의 초성 혼동행렬과 유형 1의 중성 혼동행렬을 적용한 결과가 모두 성공해야만 인식된 문자 후보가 상위 주소문자의 혼동문자라고 인정한다. 혼동행렬은 상위 주소문자가 인식된 후보 문자들과 매칭에 실패할 경우에 적용한다.



(그림 15) 혼동행렬의 적용방법

5. 실험 결과

5.1 우편봉투로부터 레이블링된 데이터를 이용한 실험결과

우편번호가 기입된 우편 봉투영상으로부터 직접 레이블링(labeling)된 광주부산의 주소 데이터 5,453개(광주 : 1,843개, 부산 : 3,610개)를 이용하여 우편번호 검증과 순로코드 발생을 실험하였다.

<표 6>에서 볼 수 있는 것처럼, 우편번호의 유효성 검증 결과 97.18%가 성공하였다. 우편번호 유효성 검증이 실패한 예는 아래와 같다.

(a) 502 - 010 : 광주 서구 서동 261 - 52 6/2
(503 - 010 : 광주 서구 서동)
(b) 500 - 091 : 광주시 북구 풍향1동 589 - 41 12/1
(500 - 090 : 광주 북구 풍향동)

위 예에서, (a)는 잘못된 우편번호 기입으로 발생한 경우이며, (b)는 우편번호가 할당되지 않은 행정동의 사용으로 발생한 경우이다.

〈표 6〉 광주부산의 레이블링된 데이터를 이용한 실험결과(5,453개)

구 분	개수	비율(%)
우편번호 유효성 검증이 성공한 경우	5,299	97.18
우편번호와 주소 상위영역 일치 검증이 성공한 경우	5,226	95.84

우편번호로 주소 사전에서 검색된 주소 단어를 이용하여 우편번호와 주소 상위영역의 일치 검증을 한 결과 95.84%의 검증 성공률을 보였으며, 이를 우편번호 유효성 검증이 된 데이터에 대해서 계산하면 검증 성공률은 98.62%가 된다. 검증되지 못한 주소의 예는 아래와 같다.

(a) 600 - 074 : 부산 중구 부평4가동 25 - 22
(600 - 074 : 부산 중구 부평동4가)
(b) 506 - 258 : 공주시 광산구 안천동 793 - 35
(500 - 090 : 광주 광산구 안청동)
(c) 500 - 070 : 부산 동구 초량3동 1196 -1번지 왕자빌딩 901호 우일선박
(주소 데이터베이스 미비로 인해 건물명 대신 번지로 순로코드 발생)

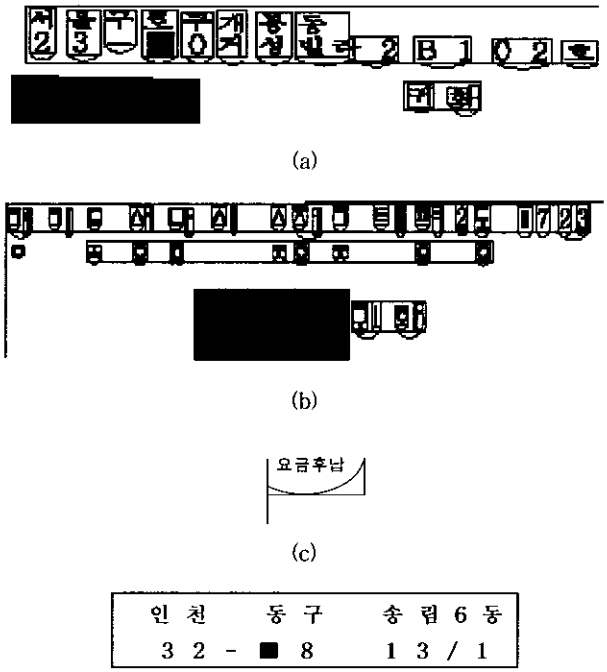
이 예에서, (a)는 잘못된 주소 기입방법으로 인해 검증을 못한 경우이며, (b)는 문자 기입 시 오류로 인해 발생하였다. 반면, (c)는 주소하위 영역을 바르게 해석하였음에도 불구하고 주소 데이터베이스의 미비로 인해 건물명 대신 주소로 순로코드를 발생한 경우이다.

실험에 사용된 DPF의 경우, 광주지역의 433개의 우편번호 중 64개를 포함하고 부산지역의 585개의 우편번호 중 282개만을 포함하고 있다. 또한, 이 DPF가 건물명에 대한 정보를 완전히 포함하고 있지 못하기 때문에 기존의 DPF가 포함하는 주소에 대해서만 순로코드를 발생하였다. 대상 주소는 전체 주소 중에서 20.08% (1095/5453)에 해당한다. 대상 주소 중 번지정보를 이용해 순로코드를 발생한 경우는 86.67%이며, 건물명 정보를 이용해 순로코드를 발생한 경우는 13.33%이다. 번지정보를 이용해 순로코드를 발생한 경우를 분석한 결과 해당 영상의 93.36%가 바른 순로코드를 발생하였다. 순로코드를 바르게 발생하지 못한 경우의 대부분은 잘못된 번지를 사용한 경우이다. 건물명 정보를 이용하여 순로코드를 발생한 경우를 분석한 결과 해당영상

의 77.40%가 바른 순로코드를 발생하였으며, 잘못된 순로코드를 발생한 경우의 대부분은 DPF가 건물명에 대한 바른 정보를 포함하지 못하기 때문이다.

5.2 인식기의 인식결과를 이용한 실험결과

우편번호와 주소 상위영역의 일치 검증 실험을 위해서 400개의 우편 영상의 인식결과를 사용했으며, (그림 16)과 같은 영상들은 실험에서 제외하였다.



(a) (b) (c) (d)

(그림 16) 실험에서 제외된 영상 (a,b) 행 분리가 잘못된 영상, (c) 주소가 없는 영상, (d) 기울어진 영상

혼동행렬의 효율성 검증을 위해 두 개의 인식기(인식기 A, 인식기 B)를 사용하였다. 인식기 B는 인식기 A보다 높은 문자 인식률을 가지며, 혼동행렬은 인식기의 인식결과를 분석하여 각각 구현하였다. 혼동행렬의 구현은 인식기 성능에 좌우되므로 인식기 A의 혼동행렬은 인식기 B의 혼동행렬보다 적용범위가 넓다. 우편번호와 주소 상위영역의 일치검증은 우편번호를 이용하여 주소사전에서 검색된 주소단어를 이용하며, 검색된 주소단어 중 마지막 계층에 해당하는 주소단어가 일치해야만 검증에 성공했다고 판단하였다. 검증 실패 시 혼동행렬을 적용해 마지막 계층에 해당하는 주소단어의 일치 여부를 확인한다. 일치에 성공하면 바로 앞 계층에 해당하는 주소단어의 일치 여부를 확인하여, 일치할 경우 검증에 성공하였다고 가정하였다.

400개의 우편영상에 대한 검증결과는 <표 7>과 같다. 혼동행렬 적용 시 인식기 A와 B는 각각 8.75%와 4.75% 검증 성공률 향상을 얻음으로 혼동행렬의 효용성을 검증할 수 있었다.

〈표 7〉 인식기의 인식결과를 이용한 우편번호와 주소 상위영역 일치 검증결과 (400개)

구 분	인식기 A		인식기 B	
혼동행렬 미사용	276	69.00%	329	82.25%
혼동행렬 사용	311	77.75%	348	87.00%

6. 결 론

본 논문은 한글주소의 해석체계를 실제 주소의 분석을 통해 제안하였으며, 제안한 주소해석방법에 근거한 주소 검색시스템의 구현에 대해 서술하였다.

우편번호가 포함하는 주소정보를 주소 상위영역으로, 나머지 주소부분을 주소 하위영역으로 구분하여 우편번호와 상위영역의 일치검증과 주소하위 영역 검증의 2단계로 순로코드를 발생한다.

주소체계를 효율적으로 표현하는 주소사전을 구성하기 위해서 6자리의 우편번호를 3개의 level로 분리했으며, 고속검색을 위해 각 level은 우편번호를 통해 직접적으로 사전구조에 접근하는 구조를 가지도록 구현하였다. 구현된 주소사전은 주소 상위영역 검증을 통해 우편번호의 유효성검증 기능도 포함하게 된다.

주소 하위영역 검증 단계는 번지정보와 건물명 정보를 이용하여 순로코드를 생성하도록 구현하였다. 번지 데이터베이스는 인식과정에서 발생 가능한 실제적인 문제를 고려한 활용방법을 제안하였으며, 번지정보와 건물명 정보가 함께 있는 경우 건물명 정보를 활용해 최종 배달점에 대한 순로코드를 발생할 수 있도록 번지 데이터베이스와 건물명 데이터 베이스를 연결시켜 구현하였다.

우편봉투 영상으로부터 레이블된 광주·부산 지역의 주소와 부분적으로 완성된 주소 하위영역 데이터베이스를 이용해 순로코드를 생성한 결과 93%의 주소에 대해 올바른 순로코드를 발생할 수 있었다. 번지정보로 순로코드를 발생시키지 못한 경우에 대해 건물명을 이용하여 순로코드를 발생하였으며, 해당주소의 77%에 대해 올바른 건물명 정보로 순로코드를 발생할 수 있었다.

또한, 우편번호와 주소 상위영역 검증과정에서 성능향상을 위해 혼동행렬을 도입하였다. 구현된 2개의 인식기에 대해 혼동행렬을 적용한 결과 각각 8.75%, 4.75% 검증 성공률 향상을 얻음으로 혼동행렬의 효용성을 확인할 수 있었다.

인쇄된 주소 영상의 해석뿐만 아니라, 문자간 혼동 가능성이 매우 큰 필기체 주소 인식과정에서 인식 후보 행정단위를 사전형식으로 제공해 줌으로써 주소 검색시스템의 유용성은 인식 성능을 속도와 정확도 측면에서 더욱 증가한다. 도시와 농촌지역을 모두 포함하는 실제 주소의 분포 및 작성 방식에 대한 통계와 전국 단위의 DPF의 구성이 완료되면, 제안한 주소 검색시스템의 고속 운영을 위해 고려해야 할 사항과 수시로 변경되는 주소 정보를 검색시스템에 쉽게 반영하는 방안에 대한 연구의 진행이 필요하다.

참 고 문 헌

[1] V. Govindaraju, A. Shekhawat, and S. N. Srihari, "Interpretation of Handwritten Addresses in US Mail Stream," International Conference on Document Analysis and Recognition, IEEE Computer Society Press, pp.291-294, 1993.

[2] S. N. Srihari and E. Kuebert, "Integration of hand-written address interpretation technology into the United States postal service remote computer reader system," Proceedings of 4th International Conference on Document Analysis and Recognition, pp.892-896, 1997.

[3] 한글 주소인식 Workshop. 한국전자통신연구원, Oct 1999.

[4] S. Madhvanath, G. Kim, and V. Govindaraju, "Chain Code Processing for Handwritten Word Recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Computer Society Press, Vol.21, No.9, pp.928-932, 1999.

[5] G. Kim and V. Govindaraju, "A Lexicon Driven Approach to Handwritten Word Recognition for Real-time Applications," IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Computer Society Press, Vol.19, No.4, pp.366-379, 1997.

[6] 김민석, 손항웅, 최완수, 김수원, "자소 추출 방법을 이용한 고속 한글인식 시스템의 구현", 전자공학회논문집(B), 제29권 제6호, pp.418-424, 1992.

[7] V. Govindaraju, E. Cohen, A. Shekhawat, and S. Srihari, "Determining the Delivery Point Code on Handwritten Addresses," Fifth Advanced Technology USPS Conference, pp.321-336, Washington D.C., 1992.

[8] W. -J. Yang, Access Schemes for Multi-Attribute-Record Structure and Color-Content-Based Image Retrieval, Ph.D. Thesis, Dept. of Electrical and Computer Engineering, State University of New York at Buffalo, NY, USA

[9] 이성환, 김은순, "주소 및 성명에서의 한글인식을 위한 효율적인 오인식 교정 알고리즘", 한국정보과학회논문지, 제20권 제5호, pp.729-738, 1993.

[10] 김수형, "최소거리 분류 및 사전기반 후처리의 강결합에 의한 필기 한글 주소열의 인식", 한국정보과학회논문지(B), 제25권 제8호, pp.1195-1205, 1998.

[11] 우편번호부. 정보통신부, May 2000.

[12] 오일석, 유태웅, 최순만, "문서영상 이진화를 위한 국부 알고리즘", 한국정보과학회논문지, 제22권 제8호, pp.1203-1212, 1995.

[13] 이성환, 김영준, "명도 문자 영상으로부터 지형적 특징 추출을 위한 효과적인 방법", 한국정보과학회논문지, 제22권 제8호, pp.1203-1212, 1995.

[14] 이진수, 권오준, 방승양, "컴퓨터가 인식하기 쉬운 한글 필기 설계", 한국정보과학회논문지, 제22권 제3호, pp.431-440, 1995.

[15] 김민기, 권오성, 권영빈, "모음의 구조적 형태와 조합 규칙에 충실한 한글 문자의 유형분류", 한국정보과학회논문지, 제25권 제4호, pp.685-695, 1998.

김 경 환

e-mail : gkim@ccs.sogang.ac.kr

1984년 서강대학교 전자공학과 졸업(공학사)

1986년 서강대학교 대학원 전자공학과 (공학석사)

1996년 미국 SUNY at Buffalo 전기 및 컴퓨터 공학과(공학박사)

1986년~1991년 금성전기(정밀) 기술연구소 선임연구원

1993년~1997년 CEDAR/SUNY at Buffalo Research Scientist

1997년~현재 서강대학교 전자공학과 조교수

관심분야 : 영상신호해석, 패턴인식, 신경회로망, Embedded System Design

이 석 구

e-mail : jajulee@samsung.co.kr

1999년 서강대학교 전자공학과 졸업(공학사)

2001년 서강대학교 대학원 전자공학과 (공학석사)

2001년~현재 삼성중공업 콘트롤시스템 연구원

관심분야 : 문자인식, 데이터베이스, RTOS, Embedded System, Control System 등

신 미 영

e-mail : shinmy@etri.re.kr

1991년 연세대학교 전산과학과 졸업(공학사)

1993년 연세대학교 대학원 전산과학과 졸업(이학석사)

1998년 미국 시라큐스대학교 대학원 컴퓨터과학과 졸업(공학박사)

1999년~현재 한국전자통신연구원 선임연구원

관심분야 : 데이터마이닝, 패턴인식

남 윤 석

e-mail : ysnam@etri.re.kr

1984년 아주대학교 산업공학과(학사)

1989년 Polytechnic Univ.(New York), Dept. of Industrial Engineering (공학석사)

1992년 Polytechnic Univ.(New York), Dept. of Industrial Engineering(공학박사)

1993년~현재 한국전자통신연구원 우정기술연구부 우정자동화팀장

관심분야 : 소프트웨어 공학, 패턴인식 등